```
impost pandas as pd
insport motplotlib . pyplot as plt
impost Seaborn as sos
 from skleam · cluster import KMeans
# Load or simulate customer data
# Sample dataset: Annual Income is spending Score
data = 3
   ( customes 1D): [1,2,3,4,5,6,7,8,9,10],
   (Annual Income (k$),: (15,16,17,18; 20,60,62,63,64,65),
  ( Spending Score (1-100)): [39,81,6,77,40,42,50,49,48,52]
3
df = pd . Data Frame (data)
# scled features for clustering
 X = of [['Annual Income (K$)', 'spending Score (1-100)']]
# Visualize data belove dustering
 Sns. scatterflot (x = (Annual Income (K$)), y = Spending Score (1-100),
  data = db)
hit . title ( customer Distribution )
 plt . show()
  Use the Elbow Method to find bitimal number of clusters
 Wcss =[]
 for i in sange (1,11):
       Kmo Kneens = KMegns (n-clusters = 1, init = K-mens++),
                         random_state = 42)
             Kmeans-fit(x)
```

```
wcss . append (Kmeans - in estia -)
hlt. plot (range (1,11), wess, monker=(6))
hlt . title ('Elbow Method for optimal Clusters')
plt . xlabel ('Number of clusters')
plt ·ylabel ('wcss')
plt. show()
# Apply K-Means with optimal clusters (eg, 3)
 Kmeans = K Means (n-dusters = 3, init = (k-means++), random_state=42)
 df[cluster] = kmeans. fit _ foredic+(x)
 # visualize clustered groups
 hit · figure (figsize = (8,5))
 ons . scatter plot (
    hue = cluster, halette = Set 2, data = df, s=100
 plt. title ('customer Segments')
 plt. show &
```

vess . offerd (Knows in ostia -) (1)= 194 ( Rushe (1) ) may be 3/4 . 7/4 Ht - title ("Elbow Method" for optional Clusters"). Output for 9 hue =

```
impost pandas as poly clustering paragraph of the thing
insport mathlotlib . hyplot as plt
 iropost Seaborn as sos
 from scipy cluster · hierarchy import linkage, dendrogram, polities
 from Sklearn . hrefmocessing import Standard Scaler Cools at a
# Sample data: Replace with real - world data as needed
 dete = { ( cluster) = colored ( linked , t=3 , coltesion = ( maxclust ) } = abb
   (Country): ['UsA), (Canada), (Snermany), (France), (India), (ching), (Brazil),
             "Russia"], (P[[ start]
 (GDP Per Capita): [63000, 46000, 48000, 41000, 2100, 12000, 8800, 11400],
(Inflation Rate': [2.3, 1.5, 1.4, 1.8, 5.5, 2.1, 4.2, 3.4],
(Unemployment Rate : [5-2,6.0,4.5,8.0,7.1,5.0,9.8,4.8] }
  d= hd. Data Frome (data)
 X = df [['GOP per Capita', 'Inflation Rate', 'Unemployment Rate']]
 # Standordize features (impostant for distance metrics)
  Scalex = Standard Scalex()
  X_scaled = scaler - fit_tonsflowm (x)
# perform heiranchical clustering using Ward's method
  linked = linkage (x-scaled, method = (word))
  # Plot dendrogram
  hlt. bigune (figsize=(10,6))
 dendrogram (linked, labels = of ("Country") . Values, orientation = (top),
   distance _sort = 'descending', show _leaf -counts = True)
```

plt. title (Dendrogram: Country Clustering Based on Economic indicators) htt. Xlobel ( country) inchest Seaboon as sons hlt. ylabel ("Com Distance") nom scipy cluster . hierarchy hopped linking hlt. tight\_layout() plt. shows . mlno & brotante troping misson of neodes ones! # optional : essign clusters - los tios sollar to show of [cluster] = 6 cluster (linked , t=3, criterian = (maxclust)) # Display clustered data ( bono) ( ASU) : ( ythus) hoint (of CC'country', cluster']]3) (000 Per Capita); [63000 218000, 48000, 24000) Polloton Rate': [203, 105, 1.4, 108, 55, 201, 4.2, 3.4], ( Voemplyment Role 1: [5-2, 6.0, 4.5, 8.0, 7.1, 5.0, 9.8, 4.3] } d= hd. Data Frome (data) X = dl [[ (GDP per Capita", "Inflation Rate", "Unenflorment Pc It standondize jesturies (infrastant jos distance moisies) Scaler = Standond Scoler() X-36led = Scales of the teans form (x) # Portoon heisenschient clustering using hand's method (inked = linkage (x-scaled, method = usend) are pashago to 19 th 1/4 - 619 ose (fines e= (10,6))

## Output for 10



